PCT

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : H04L 29/06	A2	(11) International Publication Number: **WO 97/40610** (43) International Publication Date: 30 October 1997 (30.10.97)

(21) International Application Number: PCT/CA97/00269

(22) International Filing Date: 23 April 1997 (23.04.97)

(30) Priority Data:
60/015,945   24 April 1996 (24.04.96)   US

(71) Applicant: NORTHERN TELECOM LIMITED [CA/CA];
World Trade Center of Montreal, 8th floor, 380 St. Antoine
Street West, Montreal, Quebec H2Y 3Y4 (CA).

(72) Inventors: WOOTTON, Bruce, Anthony; 10601 Bent Twig
Drive, Raleigh, NC 27613 (US). COLVIN, William, G.;
874 Childs Drive, Milton, Ontario L9T 4J6 (CA).

(74) Agent: GRANCHELLI, John, A.; Northern Telecom Limited,
Patent Dept., P.O. Box 3511, Station "C", Ottawa, Ontario
K1Y 4H7 (CA).

(81) Designated States: AU, CA, CN, JP, KR, European patent
(AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU,
MC, NL, PT, SE).

Published
*Without international search report and to be republished
upon receipt of that report.*

(54) Title: INTERNET PROTOCOL FILTER
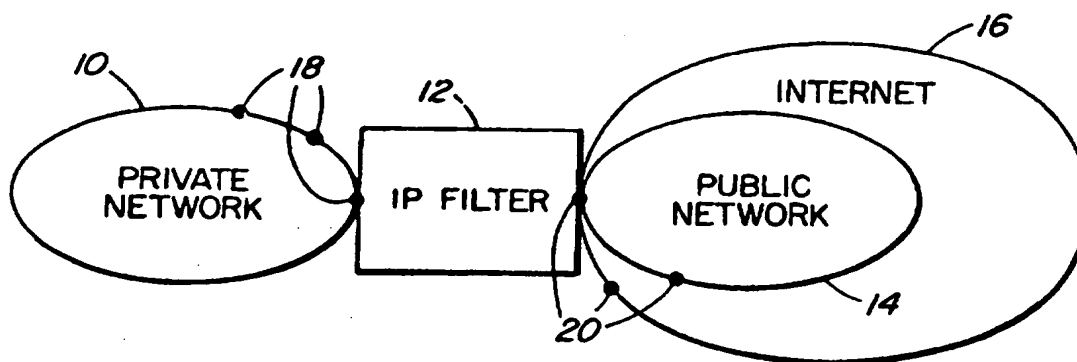
(57) Abstract

The IP filter (12), embodying the present invention, is a communications device designed to provide public network (14) or Internet
(16) access to nodes (18) of private networks (10), advantageously without requiring the private nodes on such networks to register public
Internet addresses. The IP filter presents a single IP address to the Internet and uses a plurality of IP ports to solve the problem of IP
address conservation. It initiates sessions by assigning private side IP sessions to a unique port of the IP filter's public address. The IP
filter effects a translation between a source port number for the private network and a destination port number for the public network for
communication therebetween. Benefits of the IP filter include private node security and conservation of Internet-registered addresses.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav	TM	Turkmenistan
BF	Burkina Faso	GR	Greece		Republic of Macedonia	TR	Turkey
BG	Bulgaria	HU	Hungary	ML	Mali	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MN	Mongolia	UA	Ukraine
BR	Brazil	IL	Israel	MR	Mauritania	UG	Uganda
BY	Belarus	IS	Iceland	MW	Malawi	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's	NZ	New Zealand		
CM	Cameroon		Republic of Korea	PL	Poland		
CN	China	KR	Republic of Korea	PT	Portugal		
CU	Cuba	KZ	Kazakstan	RO	Romania		
CZ	Czech Republic	LC	Saint Lucia	RU	Russian Federation		
DE	Germany	LI	Liechtenstein	SD	Sudan		
DK	Denmark	LK	Sri Lanka	SE	Sweden		
EE	Estonia	LR	Liberia	SG	Singapore		

INTERNET PROTOCOL FILTER

Background Of The Invention

The present invention generally relates to inter-
network firewalls and, in particular, to an internet
5   protocol (IP) filter whereby a private IP network domain is
mapped to a single IP address on the public Internet.

Firewalls are generally known and characterized by
computer servers which function to couple nodes within the
domain of the private network to nodes in a public network
10   domain, such as the Internet. A deficiency of the known
firewall products is the need for a unique public IP
address for each concurrent session or interaction between
public and private nodes.

A firewall providing conservation of public IP
15   addresses would be desirable.

Summary Of The Invention

It is an object of the present invention to
provide a new and improved apparatus for communicatively
coupling two networks.

20   The invention, therefore, according to a first
exemplary aspect provides a method of interfacing private
and public data communications networks, through a filter
node in communication with both networks, the filter node
having an address known in the public network, comprising
25   the steps of: routing from nodes in the private network, to
the filter node, data packets having destination
information, which includes a destination address and a
destination port, corresponding to nodes in the public
network and having source information, which includes a
30   source address and a source port, of the respective private
network nodes; for each data packet received from the
private network, at the filter node, maintaining the source
information taken from the data packet in correlation with
a unique value representing a port of the filter node, and
35   replacing in the data packet the source address with the
filter node address and the source port with the filter
node port value; and routing from the filter node, in the

public network, the data packets having the replaced source
information, according to the destination information in
each, to the corresponding public network nodes.

According to a second exemplary aspect, the
5   invention provides a method of interfacing private and
public data communications networks, through a filter node
in communication with both networks, comprising the steps
of: (a) receiving at the filter node, from the private
network, a data packet having an a destination address
10   corresponding to a node in the public network and a source
address corresponding to a node in the private network; (b)
maintaining, by the filter node, the source address taken
from the data packet; (c) replacing, in the data packet,
the source address with an address of the filter node; (d)
15   routing from the filter node, in the public network, the
data packet having the replaced source address, according
to the destination address, to the corresponding public
network node; (e) waiting for a return packet from the
public network, responsive to the data packet having the
20   replaced source information; (f) replacing, in the return
packet, the destination address with the maintained source
address; and (g) routing from the filter node, in the
private network, the return packet having the replaced
destination address to the corresponding private network
25   node.

According to a third exemplary aspect, the
invention provides a method of operating a filter node for
interfacing first and second data communications networks,
comprising the steps of: receiving from the first network,
30   a data packet having destination information, which
includes a destination address and a destination port,
corresponding to a node in the second network and having
source information, which includes a source address and a
source port, corresponding to a node in the first network;
35   maintaining the source information taken from the data
packet in correlation with a unique value representing a
port of the filter node; replacing in the data packet the

- 3 -

source address with an address of the filter node and the
source port with the filter node port value; and sending to
the second network the data packet having the replaced
source information, whereby that packet is routed according
5   to its destination information to the corresponding second
network node.

According to a fourth exemplary aspect, the
invention provides a filter node for interfacing first and
second data communications networks, comprising: means for
10 receiving from the first network, a data packet having
destination information, which includes a destination
address and a destination port, corresponding to a node in
the public network and having source information, which
includes a source address and a source port, corresponding
15 to a node in the first network; means for maintaining the
source information taken from the data packet in
correlation with a unique value representing a port of the
filter node; means for replacing in the data packet the
source address with an address of the filter node and the
20 source port with the filter node port value; and means for
sending to the second network, the data packet having the
replaced source information, whereby that packet is routed
according to its destination information to the
corresponding second network node.

25   An IP filter, embodying the present invention, is
a communications device designed to provide public network
or Internet access to nodes of private networks,
advantageously without requiring the private nodes on such
networks to register public Internet addresses. The IP
30 filter presents a single IP address to the Internet and
uses a plurality of IP ports to solve the problem of IP
address conservation. It initiates sessions by assigning
private side IP sessions to a unique port of the IP
filter's public address whereby up to 64,512 (= 65,536
35 total - 1,024 well known ports) concurrent sessions may be
supported through the single IP address. The IP filter
effects a translation between a source port number for the

- 4 -

private network and a destination port number for the
public network for communication therebetween. Benefits of
the IP filter include private node security and
conservation of Internet-registered addresses.

5   In a particular embodiment, the IP filter may
support three data transport protocols over the internet
protocol: transmission control protocol (TCP), user
datagram protocol (UDP) and Internet control message
protocol (ICMP). Packets of other protocols may be
10   ignored.
           The TCP protocol prepends a TCP header to a data
packet. The source port and destination port numbers are
contained in this header. The Internet addresses of the
source and destination nodes are contained in the IP
15   header. The IP address and port information extracted from
each packet will be used to determine where the IP filter
should route this packet.
           The IP filter maintains a lookup table of
information on each TCP connection. This information
20   includes the port from the private node, the private IP
address, the assigned port number of the destination node,
and the port number of the IP filter in the form of an
index. When a packet is received from the private network,
the private address and port number are added to the table
25   as a new entry, if an entry corresponding to this packet is
not found in the table and if the TCP header indicates that
this is a new connection request. Then the source address
and port number in the packet header are replaced with the
IP filter's IP address and port number, and the packet is
30   transmitted to the Internet.
           When the IP filter receives a packet from the
Internet, the destination port number is used to index the
lookup table. When the corresponding table entry is found,
the destination address and port number are replaced with
35   the private network's IP address and port number, and the
packet is transmitted to the private network. If the
received packet's source port is different from the port

recorded in the table, and if the packet header information
indicates that this packet is the first response on the
connection, then the lookup table is updated with the port
number assigned by the Internet node, if needed. When the
5   IP filter detects an end of transmission code in the
packet, the lookup table entry is zeroed. If the IP filter
receives packets from the Internet that do not have entries
in the lookup table corresponding to the IP filter port, it
ignores the packets.

10   The UDP protocol is connectionless, as opposed to
TCP, a connection-oriented protocol. The UDP header
contains no codes governing initial connection or end of
transmission. The data of interest in the UDP header are
the source port and destination port. This information,
15   along with the Internet addresses contained in the IP
header, are used to determine where the IP filter should
route this packet.

            The IP filter maintains a lookup table of
information on each UDP session. When the IP filter
20   receives a UDP packet from the private network, it records
the source address, the source port number, the destination
port number, and the assigned IP filter port number as the
index to the table. Then the private node address and port
number in the packet header are replaced with the address
25   and assigned port number of the IP filter. Then the packet
is transmitted to the Internet.

            When the IP filter receives a UDP packet from the
Internet, it indexes the UDP lookup table and replaces the
packet's destination information, namely the IP filter
30   address and assigned port number, with the private address
and port number from the lookup table. The lookup table
also maintains an interval indication for an expiration
timer on datagram packets received as per standard UDP
implementations. If the IP filter receives packets from
35   the Internet that do not have entries in the lookup table
corresponding to the IP filter port, it ignores the
packets.

- 6 -

As ICMP packets do not contain port numbers of either source or destination, any ICMP packets received from the private network are processed one at a time, with buffering of additional ICMP packets. The IP filter reads
5   the private address from the packet header and replaces it with the address of the IP filter. The packet is transmitted to the Internet, and the IP filter waits for the response. When it receives the responding packet, the destination address in the packet header is changed from
10   that of the IP filter to that of the node on the private network. Then the IP filter transmits the packet to the private network.

To successfully deliver packets over an IP protocol network, each node must maintain a table of other
15   hosts' IP addresses and their corresponding Ethernet addresses in an Ethernet based data communications network. The nodes actually use the IP addresses and the Ethernet addresses to address packets. The relationship between the two addresses is dynamic; that is, a node with an IP
20   address may change its Ethernet address. The information in the address table is obtained from the replies to the node's broadcast of ARP packets. The source node broadcasts ARP packets to request the Ethernet address of the destination node, given the destination node's IP
25   address. If the destination node receives the packet, it sends a reply packet with the requested information.

Though it does not maintain a true ARP table, the IP filter passes ARP packets in a manner similar to TCP and UDP packet passing. When the IP filter receives an ARP
30   packet from a node on the private network destined for the public network, it replaces the source address information with the filter's address information. The private node's IP address and the target IP address are placed in a lookup table. When the target node replies with its own Ethernet
35   address, the destination address information is changed from that of the IP filter to that of the private node before transmitting the packet to the private node. The

- 7 -

private node address information is obtained from the
table. When an ARP packet is destined for the firewall,
the ARP packet does not pass through the IP filter but is
restricted to communications between the filter and the one
5   side of the network.

Events and errors encountered by the IP filter may
be logged, for example, by writing them into a text file.

The IP filter ideally will process packets as fast
as the networks present them but when network traffic is
10   too heavy, the IP filter will then buffer the packets in
two queues, one for the private network and one for the
Internet.

Two source and destination lookup tables may be
utilized, one for TCP packets and the other for UDP
15   packets. Each table is directly indexed by the IP filter
port number assigned to the communication session. The
table entries contain the IP address of the private node,
the source port of the private node, and the destination
port of the Internet node. If there is no connection on a
20   certain IP filter port, then the corresponding entry in the
table may be zeroed. Packets arriving from both the
private network and the Internet are processed using the
same lookup table. This arrangement assumes that of the
available IP filter communications ports some are
25   designated for UDP communication and some for TCP
communication.

Brief Description Of The Drawings

The invention will be better understood from the
following description together with reference to the
30   accompanying drawings, in which:

Figure 1 is a schematic representing an internet
protocol filter coupling a private network and a public
network; and

Figure 2 is a block diagram representing internal
35   components of the filter.

Detailed Description

Referring to Figure 1, shown for illustration of

- 8 -

the present invention is a private network 10
communicatively coupled through an internet protocol (IP)
filter 12 to a public network 14 which may form part of a
global data network, otherwise referred to as the Internet

5   16.   The private network 10 represents a conventional data
communications network, such as a local area network (LAN),
having a plurality of nodes 18 each being identified by a
unique IP address within the domain of the private network
10.   The public network 14 and Internet 16 are

10   representative of public domain data communications
networks also having a plurality of nodes 20 with
corresponding IP addresses.
         The IP filter 12 acts as a gateway through which
data packets are exchanged between the private network 10

15   and the public network 14, thereby providing Internet
access to the nodes 18 of the private network 10.   The IP
filter 12 constitutes one of the private network nodes 18
and is the only such node to have a public IP address that
is Internet-registered, whereby the IP filter 12

20   essentially also constitutes one of the public nodes 20 and
its IP address is known in the public domain.   The IP
addresses of the other private network nodes 18 are
reserved for the private network 10, and not known or
registered in the public Internet address domain.   As is

25   conventional, associated with the IP address of the IP
filter 12 are a plurality of IP ports, specifically 65,536
in total of which 64,512 are not reserved for predefined
protocols and can be used for address translations.
         Communications between nodes 18 on the private

30   network 10 are unaffected by the presence of the IP filter
12, but to access the public network 14 and particularly
the nodes 20 therein, the private nodes 18 route all
communications requests through the IP filter 12.   The IP
filter 12 manages the communications between private nodes

35   18 and the Internet nodes 20 by modifying header
information of data packets received from the private
network 10 before transmitting each to the public network

14. The modifications cause the communications between the
private nodes 18 and the public Internet nodes 20 to
actually be between the IP filter 12 and the Internet nodes
20, which route all return communications to the IP filter
5   12 which subsequently routes the return data packets to the
private nodes 18.

The IP filter 12 accepts no connection requests
from the public network 14. All communications between
private nodes 18 and public nodes 20 are initiated by the
10   private nodes 18. The IP filter 12 is designed to support
three data transport protocols over the internet protocol:
TCP, UDP and ICMP messages; packets of other protocols are
rejected or ignored.

A translation table is maintained by the IP filter
15   12 to map address and ports for packets received from the
private network 10 destined to the public network 14 and
vise versa. The translation table contains the following
for each entry:

private IP address	(pIP)
private port	(pPort)
internet (public) IP address	(iIP)
internet (public) Port	(iPort)
timer	
session type/state	
Ethernet address	

20

25

The basic translation substitutes IP addresses and ports
from the private network side to the IP filter's IP address
and ports, thereby hiding all nodes 18 on the private
network 10 from the public network 14.

30   A packet originating on the private network side
specifies a source - destination of

(pIP, pPort - iIP, iPort)

This defines a "socket" in which the endpoints of the
connection (source and destination) are defined by the IP
35   addresses in the IP header and the ports in the TCP or UDP
header.

The IP filter 12 will translate the above to

- 10 -

                    (frIP, frPort - iIP, iPort)
where frIP is the IP address of the IP filter 12 on the
public network 14, and frPort is the index into the
translation table plus an offset value, for example, of
5   1024 to skip using well known ports. The frPort represents
an arbitrary port.
          The internet node 20 will reply with a packet
                    (iIP, iPort - frIP, frPort)
which will be received by the IP filter 12 and translated
10   thereby to
                    (iIP, iPort - pIP, pPort)
          In general, to translate from the private side,
the values (protocol type, pIP, pPort, iIP, iPort) must be
located in the translation table. This should be done with
15   a hash table lookup.
          Translating from the public side can be a direct
table lookup since frPort minus 1024 is the index into the
table. If (iIP, iPort) in the packet does not match the
corresponding entries in the table, then an unauthorized
20   access is logged and the packet dropped.
          In translating packets, when a port is substituted
in the TCP or UDP header, the checksum in both the TCP/UCP
and IP header must be recalculated. When an IP address is
substituted in the IP header, the IP header checksum must
25   be recalculated.
          Following are special considerations for different
protocols supported by the IP filter 12.
          In respect of TCP, when a SYN packet is received
from the private network 10, the IP filter 12 locates an
30   unused entry in the table and fills it in, setting the type
to TCP and state to SYN. Then the packet is forwarded by
the general scheme above. If no free entries exist in the
table, then the packet is dropped and the event is logged.
          If a SYN packet is received from the public
35   network 14 interface, it is treated as unauthorized and
logged (except for FTP special case described below).
However, a SYN+ACK packet is forwarded if the state of the

translation table entry is SYN. After forwarding such a
packet the state set to OPEN.

5   If a FIN packet is received by the IP filter 12
and if the state in the translation table is not FIN, the
state is set to FIN and the packet forwarded. If the state
is FIN, then the packet is forwarded and the translation
table entry is deleted by setting it to 0. A FIN must be
sent by each side to close a TCP connection.

           If a RST packet is received, then the translation
10   table entry is deleted.

           Having regard now to the UDP protocol, when any
UDP packet is received from the private network 10 side,
the IP filter 12 first tries its standard lookup. If a
translation table entry is not found, an unused entry is
15   set up and the state set to OPEN. If a free entry is not
found in the table, then rather than dropping the packet, a
random UDP in the table is overwritten. Since UDP is
connectionless and consequently an unreliable transport, if
a packet is received from the public network 14 that would
20   have needed the entry that was overwritten, that packet
will be dropped and the node 18 on the private side will
need to retry.

           With regard to FTP, an FTP client establishes a
TCP "control" connection with an FTP server on a particular
25   port, for example, port 21. However, when data is to be
transmitted, the FTP server will open a TCP connection from
its "data" port, for example, which is default 20, to a
destination port specified by the client.

           To support this, packets sent by the private
30   network 10 to port 21 need to be analyzed for an FTP "port"
command at the IP filter 12. If detected, then a new entry
in the table must be set up with pPort set to the value in
the FTP port command. The IP address and port number in
the FTP command must be changed to the IP filter's address
35   and port before forwarding the packet. The state is set to
FTPDATA.

           When a SYN packet is received from the public

- 12 -

network 14, if a table entry exists and is in FTPDATA
state, then the packet is forwarded and the state set to
OPEN.

5   For the ICMP protocol, if an ICMP packet is
received from the private network 10 and if that packet is
an echo request (ping), then the IP filter 12 locates a new
entry in the translation table. The sequence field of the
packet is stored in pPort in the table and the table index
is put in the sequence field of the packet. The ICMP
10   checksum is recalculated and the standard IP header
substitution is done. The type is set to ICMP and state to
PING and the timer set to 1 minute.

If an echo reply (ping) is received from the
public network 14 interface, then the sequence field is
15   used as the index into the table. If the state is PING,
then pPort in the table is substituted into the sequence
field of the packet, the ICMP checksum recalculated and the
standard IP header substitution is done. The table entry
is then deleted.

20   If an echo request (ping) is received from the
public network 14, then the IP filter 12 will reply. This
allows internet access to confirm that the IP filter 12 is
reachable and running.

If a Destination Unreachable packet is received
25   from the public network 14, then the header information
contained is extracted. If the protocol was TCP or UDP,
the (frIP, frPort - iIP, iPort) of the originating packet
can be determined and the translation table entry located.
If the IP address extracted from the ICMP matches the
30   address in the table, the IP filter 12 forwards the packet
to the private network 10 using the standard scheme.

All other ICMP packets received from either side
are dropped and logged.

Since most data communications protocols are based
35   on either the UDP or TCP protocols, these other protocols
are compatible with the IP filter 12 as long as they do not
initiate negotiations like FTP to have the server open a

connection back to the client. Examples of other
compatible protocols include: Telnet; TFTP (Trivial File
Transfer Protocol); DNS (Domain Name Services); and Web
browsers.

5   Whenever a packet is transmitted in either
direction, the timer field of the translation table entry
is set to the configured timeout value (except ping). Each
minute, the timer field of all active entries in the tables
are decremented and if they become 0, then the translation
10   table entry is deleted. This will clear out UDP and PING
entries which are no longer in use and also TCP entries
which have had an abnormal termination and did not send FIN
from each side. It could be a security hole to leave an
unused entry in the table for too long. A good timeout
15   value to be configured would be just longer than the
typical TCP keep alive.

           According to a particular embodiment, the private
network 10 and the public network 14 are Ethernet based
LANs. The IP filter 12 may be implemented by a data
20   processing platform which is equipped with two conventional
Ethernet hardware interfaces connected to networks 10 and
14, respectively, and which is provisioned with appropriate
software to implement the functionality of the IP filter
12.

25   Internal components of the IP filter 12 in terms
of software executable by the data processing platform are
shown in Figure 2. The internal components include two
packet drivers 30 and 32, an address resolution protocol
(ARP) table 34, an Ethernet address table 36, an IP handler
30   38, an address translation 40 and a user interface 42. The
packet drivers 30 and 32 control the Ethernet hardware
interfaces in order to communicate with, respectively, the
private network 10 and the public network 14. The IP
handler 38 provides a router functionality for receiving
35   and forwarding messages, and maintains the ARP table 34 and
the Ethernet table 36. The address translation 40 effects
translation between source port numbers from the private

network 10 and the destination port numbers on the public
network side 14. The user interface 42 enables an
operator, via a keyboard and display terminal attached to
the processing platform, to interface with the IP filter

5   12. Functions keys are provided to configure the IP
filter, view or copy log files, display status, etc. The
log file will contain the connect time of TCP or UDP
sessions, inbound and outbound traffic statistics, and
invalid access to the IP filter 12. To prevent the log

10   file from growing too large, this information will be
logged to a new file when the date changes.

Routing of packets to and from the IP filter 12 is
described in the following in terms of a public interface,
from the view of the public network 14, and of a private

15   interface, from the view of the private network 10.

The public interface behaves as a host on the LAN
segment. To forward a packet, it checks to see if the
destination IP is on the local LAN segment. If it is, it
looks up the IP address in its ARP table to find the

20   Ethernet address. If there is no entry in the ARP table,
it must put the packet on a queue and send out an ARP
request to get the Ethernet address. Standard aging out of
ARP table entries needs to be done. If the IP destination
is not on the LAN segment, it will forward the packet to

25   the configured default router. ICMP Redirect messages sent
by the default router will be ignored.

The private interface effects the functionality of
a router, as it needs to be able to forward packets to one
or more routers to communicate with the remote client

30   stations. A large remote client network may access
multiple router machines. Conventional routing can result
in large routing tables because the routing entries become
host addresses instead of subnet addresses. That is, if
the network is set up so that a client may come in through

35   either Router1 or Router2, then no single router can be the
router for the subnet that that client station is on. A
conventional router that would get routing tables via RIP

from all routers on the private network would end up with a
large table of host addresses for each remote client
connected. This can affect performance in the search time
necessary to find the route, the memory required for large
5   tables and the amount of RIP traffic on the LAN segment
between all these routers.

To handle routing in this environment, the IP
filter will maintain an Ethernet table. For every packet
that is forwarded from the private to public side, if a
10   translation entry exists, use its Ethernet index to compare
with the Ethernet source address of the incoming packet.
If they match, nothing more needs to be done. Otherwise,
the Ethernet table is searched for the source Ethernet
address, adding a new Ethernet table entry if not found.
15   The index to the Ethernet table is then saved in the
translation table entry. Then when a packet is being
translated from the public to private side, the Ethernet
address can be retrieved directly from the index in the
translation table. Thus packets will be routed to the
20   router which forwarded the packet to the IP filter.

Those skilled in the art will recognize that
various modifications and changes could be made to the
invention without departing from the spirit and scope
thereof. It should therefore be understood that the claims
25   are not to be considered as being limited to the precise
embodiments set forth above, in the absence of specific
limitations directed to each embodiment.

WE CLAIM:

1.   A method of interfacing private (10) and public
(14) data communications networks, through a filter node
(12) in communication with both networks, the filter node
5   having an address known in the public network, comprising
the steps of:
           routing from nodes (18) in the private network, to
the filter node, data packets having destination
information, which includes a destination address and a
10   destination port, corresponding to nodes (20) in the public
network and having source information, which includes a
source address and a source port, of the respective private
network nodes;
           for each data packet received from the private
15   network, at the filter node, maintaining the source
information taken from the data packet in correlation with
a unique value representing a port of the filter node, and
replacing in the data packet the source address with the
filter node address and the source port with the filter
20   node port value; and
           routing from the filter node, in the public
network, the data packets having the replaced source
information, according to the destination information in
each, to the corresponding public network nodes.
25

2.   A method as claimed in claim 1, comprising the
steps of:
           routing from nodes in the public network, to the
filter node, data packets each having the address of the
30   filter node as the destination address;
           for each data packet received from the public
network, at the filter node, correlating the destination
port of the destination information in the data packet to
particular source information being maintained and
35   replacing, in the data packet, the destination information
with the particular source information;
           routing from the filter node, in the private

network, the data packets having the replaced destination
information to the corresponding private network nodes.

3.   A method as claimed in claim 2, comprising
5   ignoring by the filter node a data packet received from the
public network, if the destination port of the destination
information in that data packet can not be correlated to
the maintained source information.

10   4.   A method as claimed in claim 3, wherein
maintaining the source information includes storing the
source information from each data packet as an entry in a
lookup table, and the filter node port value correlating to
the source information constitutes an index into the table
15   for that entry.

5.   A method as claimed in claim 4, wherein the data
packets include packets in accordance with a transmission
control protocol (TCP) over an internet protocol (IP).
20

6.   A method as claimed in claim 5, comprising
receiving at the filter node a TCP packet from the private
network; and if an entry corresponding to the TCP packet is
not found in the lookup table and the TCP packet indicates
25   that this is a connection request, storing the source
information together with the destination information from
the TCP packet as a new entry in the lookup table.

7.   A method as claimed in claim 6, comprising
30   receiving at the filter node a TCP packet from the public
network; and if the source port in the received TCP packet
is different from the destination port in a source
information entry of the lookup table, indexed by the
destination port in the TCP packet, and if the TCP packet
35   indicates that this packet is a first response to the
connection request, then updating by the filter node the
destination port in the table entry with the source port

- 18 -

from the received TCP packet.

8.   A method as claimed in claim 7, comprising
receiving at the filter node a TCP packet having an end of
5   transmission code in the packet and zeroing an entry in the
lookup table corresponding to the received TCP packet.

9.   A method as claimed in claim 4, wherein the data
packets include packets in accordance with a user datagram
10 protocol (UDP) over an internet protocol (IP).

10.   A method as claimed in claim 9, comprising
receiving at the filter node a UDP data packet from the
private network, and adding the source information and the
15 destination information from the UDP packet together with
an interval indication for an expiration timer as a new
entry in the lookup table.

11.   A method of interfacing private (10) and public
20 data (14) communications networks, through a filter node
(12) in communication with both networks, comprising the
steps of:
(a)   receiving at the filter node, from the private
network, a data packet having an a destination address
25 corresponding to a node (20) in the public network and a
source address corresponding to a node (18) in the private
network;
(b)   maintaining, by the filter node, the source
address taken from the data packet;
30 (c)   replacing, in the data packet, the source address
with an address of the filter node;
(d)   routing from the filter node, in the public
network, the data packet having the replaced source
address, according to the destination address, to the
35 corresponding public network node;
(e)   waiting for a return packet from the public
network, responsive to the data packet having the replaced

source information;

(f)   replacing, in the return packet, the destination
address with the maintained source address; and

(g)   routing from the filter node, in the private
5   network, the return packet having the replaced destination
address to the corresponding private network node.

12.   A method as claimed in claim 11, comprising
buffering, at the filter node, further data packets
10   received from the private network while waiting for the
return packet, and repeating steps (b) through (g) on an
individual basis for the further packets, if any, that were
buffered.

15   13.   A method as claimed in claim 12, wherein the data
packets include packets in accordance with an internet
control message protocol (ICMP).

14.   A method of operating a filter node (12) for
20   interfacing first (10) and second (14) data communications
networks, comprising the steps of:

        receiving from the first network, a data packet
having destination information, which includes a
destination address and a destination port, corresponding
25   to a node (20) in the second network and having source
information, which includes a source address and a source
port, corresponding to a node (18) in the first network;

        maintaining the source information taken from the
data packet in correlation with a unique value representing
30   a port of the filter node;

        replacing in the data packet the source address
with an address of the filter node and the source port with
the filter node port value; and

        sending to the second network the data packet
35   having the replaced source information, whereby that packet
is routed according to its destination information to the
corresponding second network node.

15.   A method as claimed in claim 14, comprising the
steps of:
          receiving from the second network, a data packet
5   having the address of the filter node as the destination
address;
          correlating the destination port of the
destination information in the data packet to particular
source information being maintained;
10   replacing, in the data packet, the destination
information with the particular source information;
          sending to the first network the data packet
having the replaced destination information, whereby that
packet is routed according to its destination information
15   to the corresponding first network node.

16.   A method as claimed in claim 15, comprising
ignoring a data packet received from the second network, if
the destination port of the destination information in that
20   data packet can not be correlated to the maintained source
information.

17.   A method as claimed in claim 16, wherein
maintaining the source information includes storing the
25   source information from the data packet as an entry in a
lookup table, and the filter node port value correlating to
the source information constitutes an index into the table
for that entry.

30   18.   A method as claimed in claim 17, wherein the data
packets include packets in accordance with a transmission
control protocol (TCP) over an internet protocol (IP).

19.   A method as claimed in claim 18, comprising
35   receiving a TCP packet from the first network; and if an
entry corresponding to the TCP packet is not found in the
lookup table and the TCP packet indicates that this is a

connection request, storing the source information together with the destination information from the TCP packet as a new entry in the lookup table.

5   20.   A method as claimed in claim 19, comprising receiving a TCP packet from the second network; and if the source port in the received TCP packet is different from the destination port in a source information entry of the lookup table, indexed by the destination port in the TCP

10   packet, and if the TCP packet indicates that this packet is a first response to the connection request, then updating the destination port in the table entry with the source port from the received TCP packet.

15   21.   A method as claimed in claim 20, comprising receiving a TCP packet having an end of transmission code in the packet, and zeroing an entry in the lookup table corresponding to the received TCP packet.

20   22.   A method as claimed in claim 17, wherein the data packets include packets in accordance with a user datagram protocol (UDP) over an internet protocol (IP).

23.   A method as claimed in claim 22, comprising
25   receiving a UDP data packet from the first network, and adding the source information and the destination information from the UDP packet together with an interval indication for an expiration timer as a new entry in the lookup table.

30

24.   A method of operating a filter node (14) for interfacing first (10) and second (14) data communications networks, comprising the steps of:
(a)   receiving from the first network, a data packet
35   having an a destination address corresponding to a node (20) in the second network and a source address corresponding to a node (18) in the first network;

- 22 -

(b)   maintaining the source address taken from the data
packet;

(c)   replacing, in the data packet, the source address
with an address of the filter node;

5   (d)   sending to the second network the data packet
having the replaced source address, whereby that packet is
routed to the corresponding second network node;

(e)   receiving a return packet from the second network,
responsive to the data packet having the replaced source

10   information;

(f)   replacing, in the return packet, the destination
address with the maintained source address; and

(g)   sending to the first network the return packet
having the replaced destination address, whereby that

15   packet is routed to the corresponding first network node.

25.   A method as claimed in claim 24, comprising
buffering further data packets received from the first
network while waiting for the return packet, and repeating

20   steps (b) through (g) on an individual basis for the
further packets, if any, that were buffered.

26.   A method as claimed in claim 25, wherein the data
packets include packets in accordance with an internet

25   control message protocol (ICMP).

27.   A filter node (12) for interfacing first (10) and
second (14) data communications networks, comprising:
           means for receiving from the first network, a data

30   packet having destination information, which includes a
destination address and a destination port, corresponding
to a node (20) in the second network and having source
information, which includes a source address and a source
port, corresponding to a node (18) in the first network;

35   means for maintaining the source information taken
from the data packet in correlation with a unique value
representing a port of the filter node;

means for replacing in the data packet the source address with an address of the filter node and the source port with the filter node port value; and

means for sending to the second network, the data
5   packet having the replaced source information, whereby that packet is routed according to its destination information to the corresponding second network node.

28.   A filter node as claimed in claim 27, comprising:
10   means for receiving from the second network, a data packet having the address of the filter node as the destination address;

means for correlating the destination port of the destination information in the data packet to particular
15   source information being maintained;

means for replacing, in the data packet, the destination information with the particular source information; and

means for sending to the first network the data
20   packet having the replaced destination information, whereby that packet is routed according to its destination information to the corresponding first network node.

29.   A method as claimed in claim 28, comprising means
25   for ignoring a data packet received from the second network, if the destination port of the destination information in that data packet can not be correlated to the maintained source information.

30   30.   A method as claimed in claim 29, wherein the means for maintaining the source information includes means for storing the source information from the data packet as an entry in a lookup table, and wherein the filter node port value correlating to the source information constitutes an
35   index into the table for that entry.

31.   A filter node (12) for interfacing first (10) and

- 24 -

second (14) data communications networks, comprising:

(a)   means for receiving from the first network, a data packet having an a destination address corresponding to a node (20) in the second network and a source address
5   corresponding to a node (18) in the first network;

(b)   means for maintaining the source address taken from the data packet;

(c)   means for replacing, in the data packet, the source address with an address of the filter node;
10   (d)   means for sending to the second network the data packet having the replaced source address, whereby that packet is routed to the corresponding second network node;

(e)   means for receiving a return packet from the second network, responsive to the data packet having the
15   replaced source information;

(f)   means for replacing, in the return packet, the destination address with the maintained source address; and

(g)   means for sending to the first network the return packet having the replaced destination address, whereby
20   that packet is routed to the corresponding first network node.

32.   A filter node as claimed in claim 31, comprising means for buffering further data packets received from the
25   first network while waiting for the return packet, and means for controlling means (b) through (g) on an individual basis for processing the further packets, if any, that were buffered.
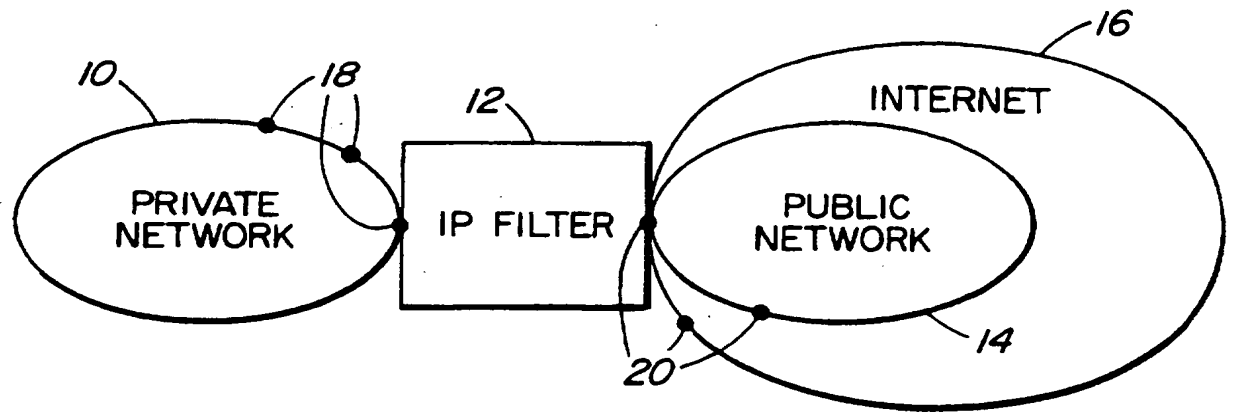
FIG. 1

FIG. 2
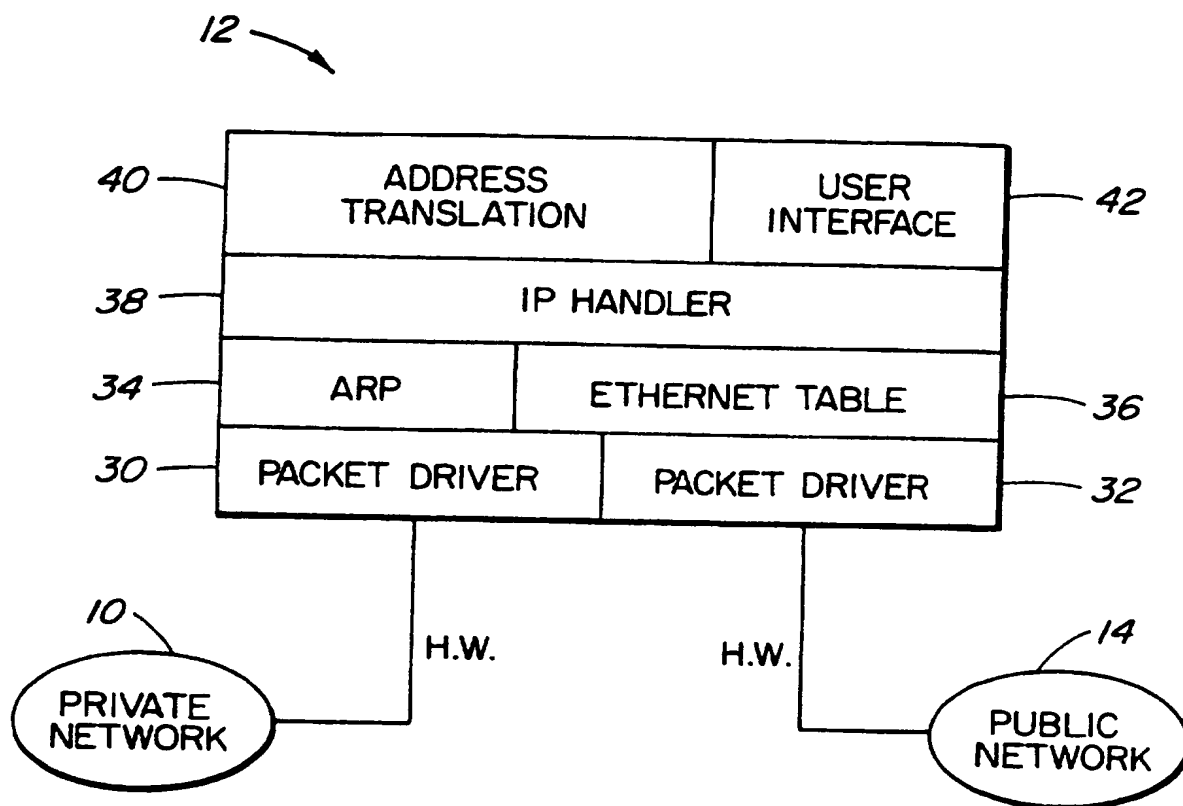
A. CLASSIFICATION OF SUBJECT MATTER
IPC 6   H04L29/06   H04L12/46

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 6   H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	RFC1631, May 1994, INTERNET ENGINEERING TASK FORCE, USA, pages 1-10, XP002040992 EGEVANG K AND FRANCIS P: "The IP Network Address Translator (NAT)" see paragraph 2; figures 1,2 see paragraph 3.3 ---	1,11,14, 24,27,31
A	EP 0 465 201 A (DIGITAL EQUIPMENT CORP) 8 January 1992 see column 7, line 30 - column 8, line 27 see column 10, line 45 - column 12, line 22; figure 2 --- -/--	1,11,14, 24,27,31

| X | Further documents are listed in the continuation of box C. | X | Patent family members are listed in annex. |

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

| Date of the actual completion of the international search | Date of mailing of the international search report |
| 19 September 1997 | 14.10.1997 |

| Name and mailing address of the ISA
European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
Fax (+31-70) 340-3016 | Authorized officer

Dupuis, H |

Form PCT/ISA/210 (second sheet) (July 1992)

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	INTERNET SECURITY HANDBOOK, 1995, MAIDENHEAD,ENGLAND, pages 27-37, XP002040993 STALLINGS W: see page 31; figure 3.2 -----	11,24,31

11

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 0465201 A	08-01-92	US 5309437 A	03-05-94
		CA 2044363 A	30-12-91
		DE 69122439 D	07-11-96
		DE 69122439 T	15-05-97

THIS PAGE BLANK (USPTO)